IN THE CLAIMS:

Complete claim listing:

1. (Original) A method for allocating entries in a branch target buffer (BTB) in a pipelined data processing system, comprising:

    fetching instructions from a plurality of instructions;

    determining that one of the plurality of instructions is a branch instruction;

    decoding the branch instruction to determine a branch target address;

    determining if the branch target address location can be obtained without causing a further stall condition in the pipelined data processing system; and

    selectively allocating a BTB entry based on the determination.

2. (Original) The method of claim 1 wherein determining if the branch target address location can be obtained further comprises examining a predetermined slot of a prefetch buffer having a plurality of slots to identify the branch instruction.

3. (Original) The method of claim 2 further comprising loading a branch target address corresponding to the branch instruction into a predetermined entry of the BTB.

4. (Original) The method of claim 2 wherein the predetermined slot of the prefetch buffer is characterized as being a first slot.

5. (Canceled)

6. (Canceled)

7. (Original) The method of claim 1 further comprising:

    determining that a stall condition exists in the data processing system;

    determining that a BTB entry will not be allocated because of the stall condition; and

    waiting for the branch instruction to be fetched from a memory location.

2

8. (Original) The method of claim 1 further comprising using a first-in, first-out replacement algorithm to load the BTB.

9. (Original) The method of claim 1 wherein determining further comprises:
   determining that the branch instruction was not loaded into a predetermined slot of a prefetch buffer and no other stall condition will occur.

10. (Original) A method for allocating entries in a branch target buffer (BTB) in a pipelined data processing system, comprising:
   fetching instructions from a plurality of instructions;
   determining that one of the plurality of instructions is a branch instruction;
   decoding the branch instruction to determine a branch target address;
   determining that the branch instruction was not loaded into a predetermined slot of a prefetch buffer and no other stall condition will occur; and
   selectively allocating a BTB entry based on the determination.

11. (Original) The method of claim 10 further comprising loading a branch target address corresponding to the branch instruction into a predetermined entry of the BTB.

12. (Original) The method of claim 10 wherein the predetermined slot of the prefetch buffer is characterized as being a first slot.

13. (Original) The method of claim 10 wherein determining further comprises:
   examining a predetermined slot of a prefetch buffer having a plurality of slots to identify the branch instruction; and
   calculating the branch target address using a program counter value and a displacement field value of the branch instruction.

14. (Original) The method of claim 13 further comprising using the branch target address in the BTB entry to prefetch a target instruction.

3

15. (Original) The method of claim 10 further comprising using a first-in, first-out replacement algorithm to load the BTB.

16. (Original) A data processing system comprising:
a prefetch buffer having a plurality of slots for storing instructions;
a branch target buffer (BTB) having a plurality of entries;
a control logic unit coupled to the prefetch buffer and to the BTB, wherein the control logic unit causes an entry of the plurality of entries of the BTB to be allocated to receive a branch target address of a branch instruction if the branch instruction is not detected in a predetermined slot of the plurality of slots of the prefetch buffer.

17. (Original) The data processing system of claim 16, wherein the predetermined slot of the plurality of slots is characterized as being a first slot having an output coupled to an input of an instruction register.

18. (Original) The data processing system of claim 16, wherein a first-in, first-out replacement algorithm is used to load the BTB.

19. (Original) The data processing system of claim 16, further comprising a multiplexer having a plurality of inputs, an input of the plurality of inputs coupled to one of the plurality of entries of the BTB, and an output for providing a selected one of plurality of entries in response to a control signal.

20. (Original) The data processing system of claim 19, further comprising a comparator having a first input for receiving an address, a second input coupled to each of the plurality of entries of the BTB, and an output for providing the control signal to the multiplexer.

21. (Original) The data processing system of claim 16, further comprising:
a program counter; and

4

a branch address calculator for calculating a branch target address of the branch
instruction using a value from the program counter and a displacement
field of the branch instruction.

22. (Original) The data processing system of claim 16, wherein the data processing system is a
pipelined data processing system implemented on a single integrated circuit.

23. (Original) The data processing system of claim 16, wherein each entry of the plurality of
entries of the BTB includes a first bit field for storing an address value related to the
address of the branch instruction and a second bit field for storing a branch target address.

24. (Original) The data processing system of claim 16, wherein the predetermined slot of the
plurality of slots is characterized as being a second slot having an output coupled to a first
slot characterized as not being coupled to an input of an instruction register.

5